ADAPTIVE METHODS

5860 Trinity Parkway, Suite 200 ■ Centreville, VA 20120 ■ ph 703-968-8040 ■ fax 703-968-8101

10 January 2013

CDR Joseph Cohn
ONR: 341
875 North Randolph St
Arlington, VA 22203-1995

Re: Contract Number: N00014-12-M-0140

Enclosures: Progress Report CLIN 0005, CDRL A002

Dear CDR Cohn,

Enclosed please find a copy of Adaptive Methods Progress Final Report as required by the referenced contract. Distribution is made via email to all parties.

If you have any questions of a technical nature, please contact Lewis Hart at 703-968-8040 ext. 320. Contractual questions should be referred to the undersigned.

Sincerely,

Michelle Garrison

Digitally signed by Michelle Garrison
DN: cn=Michelle Garrison, o=Adaptive Methods, Inc., ou, email=mgarrison@adaptivemethods.com, c=US
Date: 2013.01.10 14:40:54 -05'00'

Michelle Garrison
Contracts Administrator
Email: mgarrison@adaptivemethods.com

Cc: Defense Technical Information Center
Director, Naval Research Lab
Sharon Campbell

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **01 OCT 2013**	2. REPORT TYPE	3. DATES COVERED **01-08-2012 to 31-12-2012**

4. TITLE AND SUBTITLE **PhysicsFun4k24**	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Adaptive Methods Inc.,5860 Trinity Parkway, Suite 200,Centreville,VA,20120**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR’S ACRONYM(S)
	11. SPONSOR/MONITOR’S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**21**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)* 01/10/2013	**2. REPORT TYPE** Final Report	**3. DATES COVERED** *(From - To)* August 1, 2012 - December 31, 2012

4. TITLE AND SUBTITLE
PhysicsFun4k24

5a. CONTRACT NUMBER
N00014-12-M-0140

5b. GRANT NUMBER
N/A

5c. PROGRAM ELEMENT NUMBER
N/A

6. AUTHOR(S)
Lewis Hart

5d. PROJECT NUMBER
N/A

5e. TASK NUMBER
N/A

5f. WORK UNIT NUMBER
N/A

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Adaptive Methods Inc.
5860 Trinity Parkway, Suite 200
Centreville, VA 20120

8. PERFORMING ORGANIZATION REPORT NUMBER
N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Office of Naval Research
875 North Randolph St.
Arlington, VA 22203-1995

10. SPONSOR/MONITOR'S ACRONYM(S)
N/A

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
N/A

12. DISTRIBUTION/AVAILABILITY STATEMENT
"Distribution Statement A: Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT
The PhysicsFun4k24 project objective was making physics fun through discovery learning. Our approach was to investigate a methodology and environment which would capture a student's interest providing an engaging story and educational value mapping curriculum into that story. Both the story and curriculum are realized in a virtual world with which the student can interact. The proof of concept was demonstrated by formulating a gaming environment for learning a basic set of physics principles where students are fully "immersed" in 3D virtual environment and enabled to discovery through experimentation.

15. SUBJECT TERMS
Discovery Learning, Virtual World, STEM, Gesture Interfaces, 3D Visualizations.

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Lewis L Hart
					19b. TELEPHONE NUMBER *(Include area code)*
U	U	U	SAR	19	703 667 7816

Standard Form 298 (Rev. 8/98)
Prescribed by ANSI Std. Z39.18

PhysicsFun4K24
Final Report

Contract N0001412M0140 - CLIN A0002
11 January 2013

ADAPTIVE METHODS, INC.

5860 TRINITY PARKWAY, SUITE 200

CENTREVILLE, VIRGINIA 20120

Technical Contact	**Business Contact**
Lewis L Hart	Judy Barhorst
Principal Investigator	Director of Contracts
lhart@adaptivemtehods.com	jbarhorst@adaptivemethods.com
703 667 7816	703 968 8040 ext 110

Contents

1 Introduction

The overarching objective of this project was, as stated in original project description, was:

> *"Making Physics Fun Through Immersive Discovery Learning to provide students with immersive and interactive tools for exploring the fundamental laws of physics and how they impact the world around us."*

Our approach was to investigate a methodology and environment which would capture a student's interest by providing an engaging story and concurrently provide educational value by mapping curriculum elements into the fabric of that story. Both the story and the curriculum are realized in the creation of a seedling game in a virtual world with which the student can interact.

1.1 Project Objectives and Requirements

The PhysicsFun4k24, or simply PhysicsFun, project defined the following objectives:

- ***Formulate a gaming environment*** for learning a basic set of physics principles where students are fully "immersed" in lessons, enabled to discovery principles and allowed to identify limits and extension of principles through experimentation;
- ***Create a prototype*** of that environment for students to experientially explore and understand a basic set of physics principles, where students interact with the virtual world in a multi-sensory manner, and
- ***Provide assessments*** embedded into the gaming environment that examines students' attitudes, confidence, and knowledge of physics and the effectiveness of the immersive gaming environment.

Beyond the satisfaction of these objectives, there are certain requirements that must be met in order for the concepts developed to be successfully deployed and used in early elementary STEM education:

- Present an ***affordable*** system that employs ***state-of-the-art*** 3D input devices and immersive gaming technology and leverages recent advances in computing systems, visual displays, and computer-generated imagery software.

- Create software environment designed to ***operate on a range of hardware*** configurations including laptop, PC or tabletop workstations and ***positioned to evolve with changing technology***.
- Provide ***curriculum coordinated with standards*** and allow that curriculum to be ***easily updated and expanded.***

1.2 Achievements

The PhysicsFun project has laid the ground work toward each of the projects objectives by putting in place foundational elements to research and build complete systems in future work. We have applied a systematic approach to developing **Immersive STEM Methodology**[1] by mapping a small set of state standard early elementary school curriculum to elements of a seedling game. This **seedling game**[2]**,** provides a gesture based interface and 3D visualization environment that casual observations during development shows resulted in an engaging, immersive environment. The concepts of gesture based interfaces were proven using the Microsoft Kinect and ASUS Xtion devices and the immersive engagement demonstrated for 3D display technologies. We believe that the work done on PhysicsFun serves to show the feasibility of the methodology and the of a immersive virtual environment.

2 Immersive STEM Methodology

The Immersive STEM Methodology, show in Figure 2-1, and further discussed in our earlier report, provided an effective approach to mapping a science curriculum space into a story space and ultimately into a virtual world implementation.

The curriculum space provided grounding and rationale for the information being learned and focused game objectives and scope. The story space provided a compelling situation and

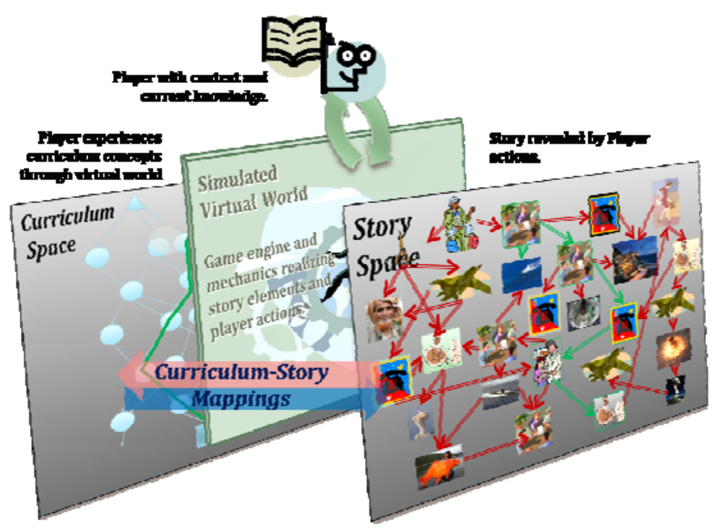


Figure 2-1 - Adaptive Methods' Immersive STEM concept, the student must apply current knowledge of a defined curriculum to succeed in the virtual world; knowledge that the student lacks can be gained while participating in the associated story presented by the virtual world.

[1] CLIN A0002
[2] CLIN A0003, CLIN A0004, and CLIN A0005

scenario based environment which gave the curriculum a “real world” application. We found the Unity3D gaming engine provides appropriate mechanics and visualization of sufficient quality and performance to engage students who have had their expectations of gaming quality set quite high.

3 The Seedling Game

This section describes the concept, design and implementation of the seedling game, which we have called “Tadpoles”.

3.1 Story Space– Navy Leap Frogs

Tadpoles presents the opportunity for students to learn what it takes to be a member of the U.S. Parachute Team “Navy Leap Frogs”[3], and gain an understanding of physics along the way. During the game students will experience the effects of gravity and aerodynamic drag on a series of virtual skydiving training missions. They learn what terminal velocity means, how to maximize their velocity and control maneuvers by understanding the relationships between air speed and their own body positions.

First Grade Topics

Force, Motion, and Energy

1.2 The student will investigate and understand that moving objects exhibit different kinds of motion. Key concepts include

a) objects may have straight, circular, and back-and-forth motions;
b) objects may vibrate and produce sound;
c) pushes or pulls can change the movement of an object; and
d) the motion of objects may be observed in toys and in playground activities.

Fourth Grade Topics

Force, Motion, and Energy

4.2 The student will investigate and understand characteristics and interaction of moving objects. Key concepts include

a) motion is described by an object’s direction and speed;
b) forces cause changes in motion;
c) friction is a force that opposes motion; and
d) moving objects have kinetic energy.

Figure 3-1 - Curriculum covered in the Tadpoles game.

[3] http://www.leapfrogs.navy.mil/

3.2 Curriculum Space - Force and Motion

Tadpoles covers a small section of physics curriculum, Figure 3-1, based on the State of Virginia Science Standards of Learning for First and Fourth grades, shown below. The modules will introduce students to Newton's first and second laws of motion. Forces are introduced using gravity, things fall toward the earth, and drag, things moving through the air produce friction which resists motion.

3.3 Game Space - Virtual World of "Tadpoles"

The above three elements, story, curriculum and virtual world, are combined to produce the "Tadpoles" game which consist of one curriculum module, with four lessons, as shown in Figure 3-2, each lesson is mapped to a single level in the game.

- Level 0 – Briefing -- Provides the Student an opportunity to learn the basic concepts of the curriculum before or after trying the gaming levels.
- Level 1 – Free Falling -- You need to be able to move fast. You need to determine how to reach your greatest speed during your decent. You can get faster by getting smaller.
- Level 2 –Landing Safe -- Introduces player to the forces of air resistance depending on overall shape (area) and how a parachute lowers terminal velocity.
- Level 3 – Hitting the Mark -- Leap Frogs must be able to land where they want to land, so they need to know how to hit the mark. Changing your shape can change which way air is pushing you, so you can steer with you arms.

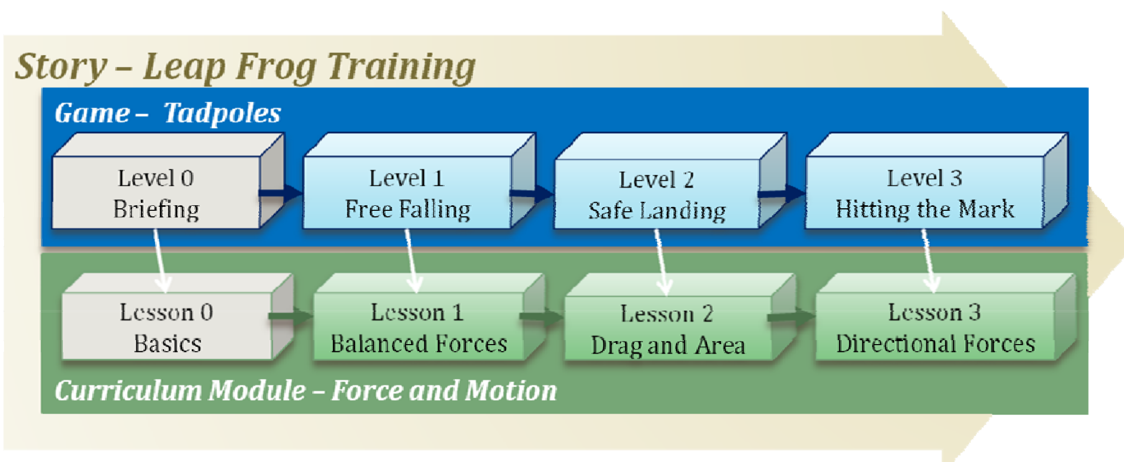


Figure 3.2 – The story provides a simple linear progression of game level mapped to a single physiscs module with four lessons.

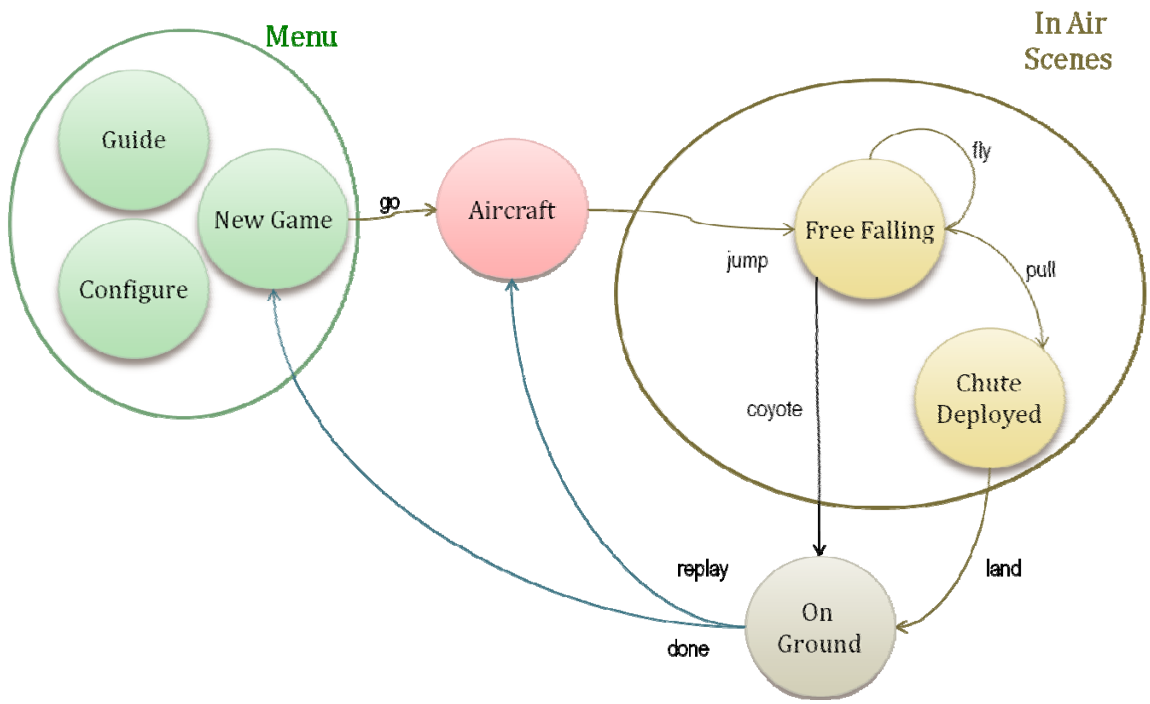


Figure 3.3 – The scenes in the game as states or modes with the possible events that cause transitions between them.

This is a very simple curriculum-to-game mapping that shows proof-of-concept. The students interact with and progress through the game using gesture and motion based APIs with both worn gloves and position tracking component as options. They are presented with a 3D presentation of the virtual world.

The game is organized internally as a series of different scenes, shown conceptually in Figure 3-3. These scenes in the game represent one or more states or modes in the game play. The green scenes are primarily administrative, the blue are help and practice, while the yellow are actual game play. Note that the game play scenes are used slightly differently for each of levels one, two and three. The arcs represent events, and the associated transitions between scenes and states. They may be generated either internally in game, such a "land" when the student avatar reaches the ground, or produced externally by gestures, such as "help", "jump" or "pull". Figures 4-3, provides a sampling of art from the game illustrating how these levels appear to the player.

Figure 3-4 - Scenes from the Tadpoles game, informations from the Survival Guide briefing, awaintg the jump in the aircraft interior, free falling in "spread Eagle and in "Dive Position", and on ground after level completion.

4 Measurement and Assessment

In order to provide data for longer term, broader analysis of student comprehension and engagement it is necessary to provide a systematic recording of students' behavior in game into persistent log. Each measure that is recorded must be well designed and meet a minimum set of requirements to support rigor in subsequent analysis.

PhysicsFun implements an in-process game measure engine. The engine measures and records this information on a per Player basis for assessment review. The measures that are recorded in PhysicsFun include:

- Who the student is – id, school, etc;
- What the student did – direct actions, not inference by game results;
- When events occurred in game and real time -- level completion, mistakes made, self-corrections, etc;
- Where (contextually) in the game the event occurred – level, scenario, specifics of play, etc.

In some cases, the measures lack sufficient Context, in that they do not record all relevant game data, or do not yet have all details, for example students are presented solely by user ID.

The measure logs may be used to calculate aggregate statistics, such as average time required completing a game level, number of mistakes made, and number of self-corrections made; to provide an assessment of satisfaction of research objectives. An example of the recorded measures and their interpretation is shown in Figure 4-1.

Type	When	User ID	Event	ID	ID2	Data
UserEvent	1356733367518	1		1		
PoseEvent	1356733368050	1		65	A	69
PoseEvent	1356733368050	1		68	D	60
PoseEvent	1356733368050	1		69	E	81
PoseEvent	1356733368050	1		81	Q	92
PoseEvent	1356733369159	1		65	A	69
PoseEvent	1356733369159	1		68	D	12
PoseEvent	1356733369159	1		69	E	16
PoseEvent	1356733369159	1		81	Q	92
PoseEvent	1356733369738	1		65	A	63
PoseEvent	1356733369738	1		68	D	-12
PoseEvent	1356733369738	1		69	E	-16
PoseEvent	1356733369738	1		81	Q	85
PoseEvent	1356733369847	1		65	A	C
PoseEvent	1356733369847	1		68	D	C
PoseEvent	1356733369847	1		69	E	C
PoseEvent	1356733369847	1		81	Q	C
PoseEvent	1356733369956	1		65	A	C
PoseEvent	1356733369956	1		68	D	C
PoseEvent	1356733369956	1		69	E	C
PoseEvent	1356733369956	1		81	Q	C
PoseEvent	1356733373598	1		32		C
PoseEvent	1356733374333	1		10		

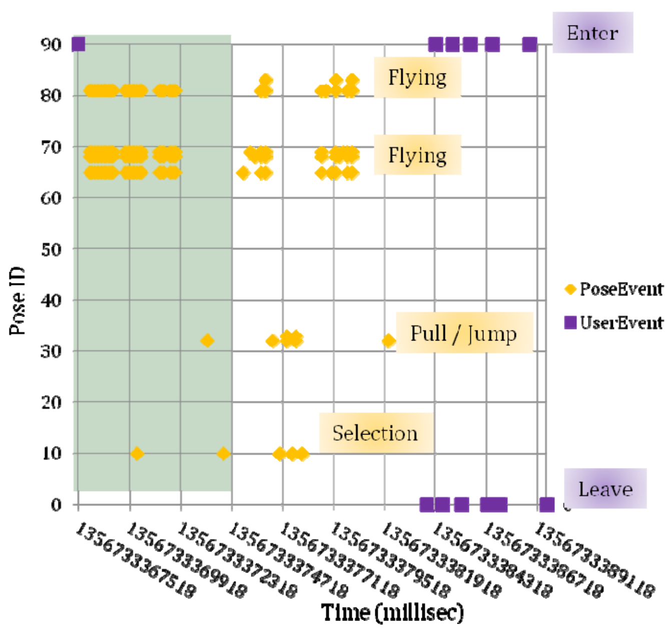


Figure 4.1 -Example metrics recorded during Tadpoles game play.

Teacher evaluation is a combination of both completion assessment and in-process assessment. In PhysicsFun there is no direct mechanism for teacher evaluation, however, during gameplay a teacher may be call upon to guide the students in understanding curriculum concepts during guided replay of levels.

5 Interface Concepts

The principal interface for PhysicsFun is Gesture based. There has been a significant amount of prior research into the types and effectiveness of gesture based interfaces (User-Defined Gestures for Surface Computing, April 2009) (Karam, November 2006) (A taxonomy of Gestures in Human Computer Interaction, 2005). We developed a basic taxonomy, shown in Figure 5-1, of user interface actions. This taxonomy was used to guide the development of the PhysicsFun interface and to identifying potential alternative approaches. The concepts which were actually used in the Tadpoles game are highlighted with red boarders, and are enumerated with additional details in Figure 5-2 below.

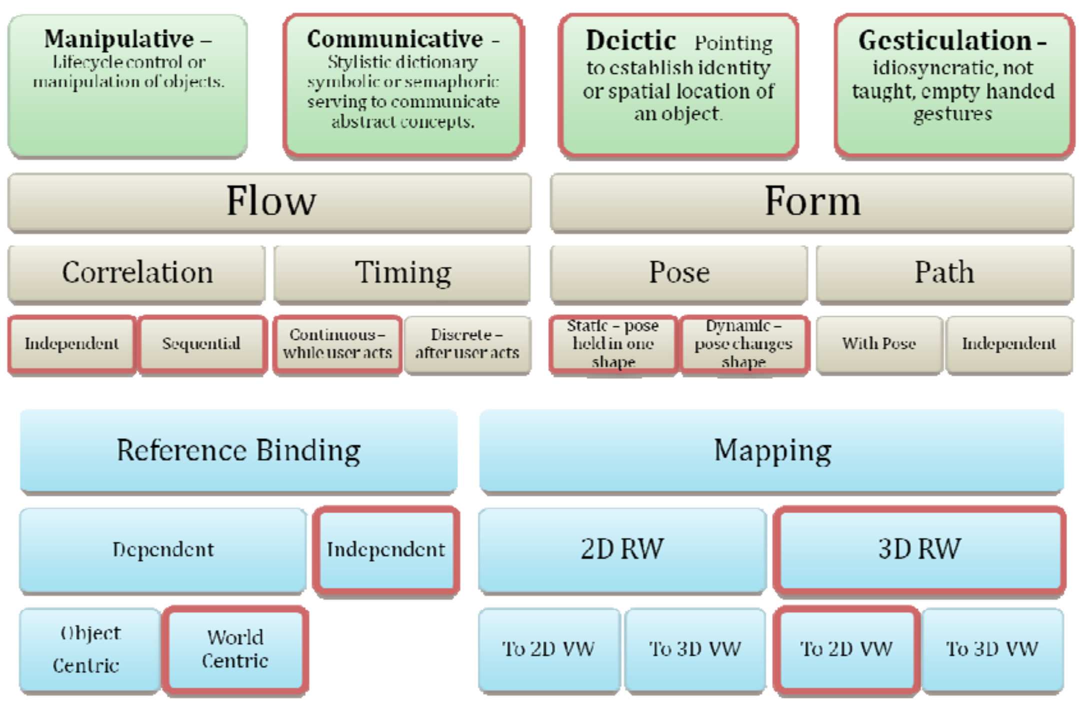


Figure 5.1 – Our simple taxonomy of gesture types and characteristics provides aguide to further investigation of interface approaches.

Pose Event	*Gesture*				*Key*
	Type	*Flow*	*Form*	*Implementation*	
Push	Deictic / Communicative	Independent / Discrete	Static	Move Right Hand and hover over desired selection on screen.	Enter
Arm/Leg Extension	Manipulative / Gesticulation	Independent / Continuous	Dynamic	The arm and leg motions used to direct the avatars flight.	Q, E|P, A|L, D
Jump	Communicative	Sequential / Discrete	Static	Begin play by moving both hands from over head to waist level.	Space
Dive	Manipulative / Gesticulation	Independent / Discrete	Dynamic	Tuck arms and legs to maximize dive speed.	S, W toggle
Pull	Communicative	Sequential / Discrete	Static	Open the parachute by moving both hands from over head to waist level.	Space
Land	N/A	N/A	N/A	Automatic in game event.	N/A

Figure 5-2 Gestures and Controls

Two basic types of gesture events have been identified. Game Control gestures which are primarily communicative, discrete and static (shaded blue), and Game Play gestures which are primarily manipulative, continuous and dynamic (shaded orange).

The Game Play gestures represent a more complex and diverse set. Three gestures, *jump, fly* and *pull*, are those which the student uses to manipulate their avatar in game, such as jumping out of the aircraft to begin. These will be based on gesticulation, using the natural motion analogous to the desired avatar action; that is the student would jump up to gesture *jump* or reach over head and then pull down to gesture *pull*. In these two cases the gestures are recognized as discrete events. The *fly* gesture, while perhaps not truly a gesticulation to the students, would be using typical sky diving positions to control speed and direction. This gesture is continuous, monitoring the student's body position during game play.

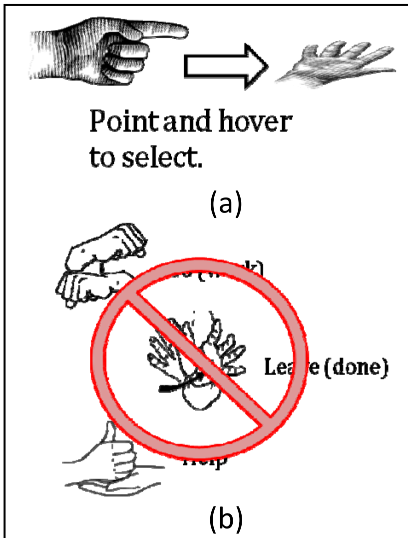


Figure 5-3 – Gestures implemented for Game Control; menu selection using point and hover (a) standalone signs we not feasible due to recognition device limitations.(b).

5.1 Keyboard and Mouse Command

While the focus of the PhysicsFun project is on immersive environments, we believe that a greater range of students may be engaged by supporting traditional keyboard, mouse or gamepad commands as well. To the extent possible, key board equivalent commands were implemented for each gesture, and are included in the table in Figure 5-2

Providing a keyboard and mouse interface provided two additional benefits, in addition to providing familiar interface to students. First, this interface provided a valuable capability during the development process, allow testing and trial play on development systems without additional hardware; and secondly, this interface method is amenable to deployment as a web-based version.

6 System Architecture

PhysicsFun combines special purpose input hardware and computer gaming software to provide an overall system archetecture shown in Figure 6-1.

6.1 Hardware Architecture

The core hardware used for PhysicsFun is a commercial off the shelf Dell T1600 Fixed Workstation equipped with a quad-core Xeon processor running Windows 7 Ultimate. It has been augmented to support 3D graphics with an ASUS ENGTX560 graphics card, ASUS 27" 120Hz LCD display and Nvidia active 3D glasses.

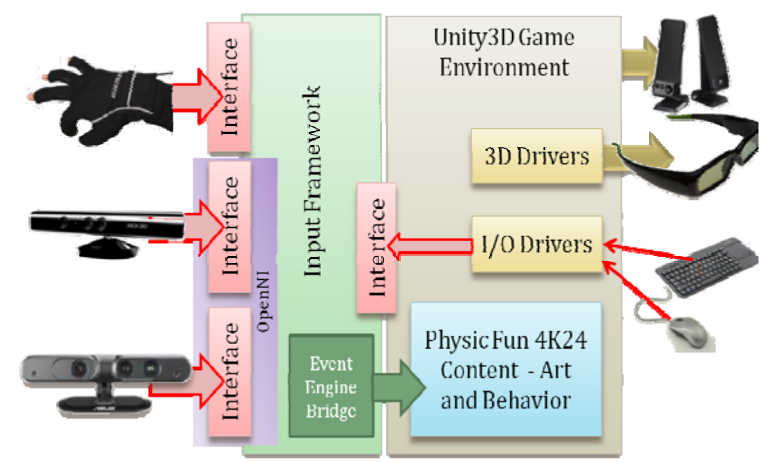


Figure 6-1 – High level system architecture for PhysicsFun -

Three gesture-based input devices were investigated, AcceleGlove and two camera base system MicroSoft Kinect and ASUS Xtion.

AcceleGloves[4] are gloves with six embedded sensors per hand to provide hand and finger acceleration measurements. Unfortunately they proved to be infeasible for two primary reasons. Since the glove contained only accelerometers, the gravity cause a bias in each sensors x,y and z axises which depended on orientation of the hand which could

[4] Production of the AcceleGlove was halted in June of 2012 and they are no longer available or supported.

Distribution Statement A - Approved for public release; distribution is unlimited.

not be disambiguated from linear acceleration cause by the hands linear movement. The inclusion of a micro-gyroscope component in the glove would allow the accelerations to be disambiguated. Secondly, the noise in the accelerometer reading, when integrate to get velocities and positions resulted in large drift in position estimates; using a low-pass filter reduces the noise but introduces a significant lag in glove response (2010).

Microsoft Kinect and ASUS Xtion PRO are very similar devices which use an IR depth measurement to produce a body skeleton tracking in three dimensions. The Kinect[5] also includes a RGB video camera and audio capability with directional sound/voice recognition, but these were not used in this project. Both of these devices provided excellent player tracking. Note however, that the ASUS devices are not well documented.

Each device had associated drivers and SDKs which provide the basic hardware interface.

6.2 Software Architecture

The software architecture has two principal components. The hardware devices are mediated through an Input Framework which isolates the hardware details from game implementation. The second component is the Gaming Environment in which the actual game art and behaviors are developed and deployed.

6.2.1 Input Framework

The input device hardware was connected to the game engine through an Input Framework. The framework provided device agnostic input streams and recognition of use gestures and actions. The data processing flow of the Input Framework is shown in Figure 6-2.

[5] ASUS release the Xtion Pro Live which provides RGB video and audio capability during the executre of this project, too late to be investigated.

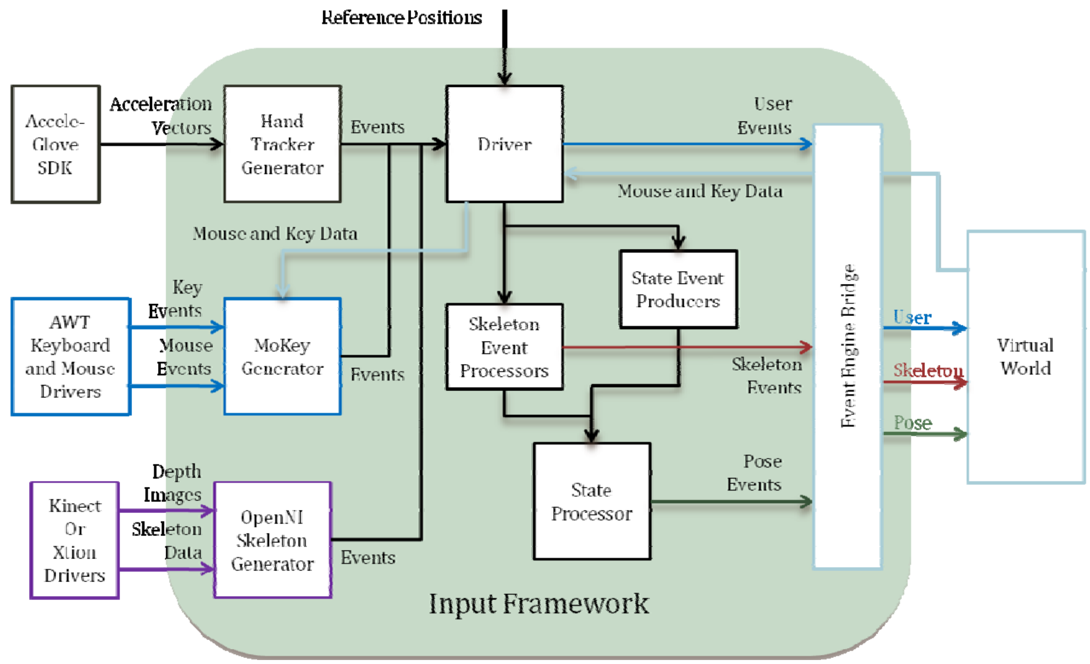


Figure 6-2 – The Input Framework provided a common interface to hardware devices which allow them to share a common event process capability.

6.2.1.1 Hardware Interfaces

Each hardware devices was interfaced into the Input Framework using an Event Generator. The Event Generator understood the details of the device, read the raw data and performed appropriate processing to produce a stream of Events. Three event types were produced:

- User Events – occurred when the player enters or leaves the gameplay area,
- Skeleton Event – occur in a continuous stream, as the player moves the 3D locations of 15 identified joints on the players skeleton (e.g. location of Left Elbow) are tracked,
- Pose Event – occur when a pose or gesture is recognized.

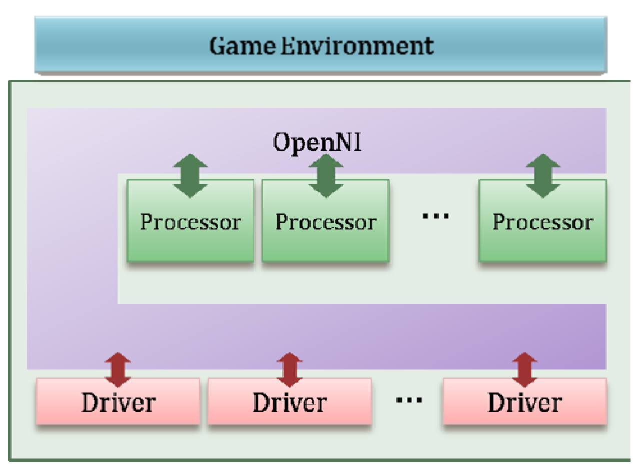


Figure 6-3 – A generic architecture based on OpenNI allows the inclusion and evolution of hardware and software components.

Not all devices produce all or complete events, but the minimum was the production of at least partial skeletons in skeleton events.

The AcceleGlove used the drivers and SDK provided with the gloves. These interfaces used USB serial connections to provide a stream of accelerometer readings from the gloves. A software interface was designed and implemented which mapped the left and right glove to left and right hand positions in the user skeleton. The other joint locations were given reasonable static values. The AcceleGloves proved incapable of reliably producing gestures or stable skeletons; User Events are generating when the gloves were connected and disconnected from the system.

The Kinect and Xtion devices used the OpenNI [6] architecture as shown in Figure 6-3, for drivers and as the SDK. OpenNI is an open source project sponsored by a number of private companies [7] interested in open gesture-based interfaces. This project used a skeleton production process to provide user skeleton tracking as the primary input. These devices provide User Events when a user was detected and lost; and provided full Skeleton Events at a rate of 30+ skeleton per second. There was no pose / gesture recognition in these generators.

The keyboard and mouse used two sources. For test, a "stickman" skeleton associated with a Java JFrame was implemented; essentially using the Java AWT as the SDK. The Unity environment was also enabled to provided mouse and key data when the game had focus by passing event data to the input framework. The Keyboard-Mouse generator produces User Events when focus is received and lost. Mouse and Keyboard AWT events were mapped into limited (not all joints moved) skeleton events and pose events. For both sources, the pose and skeleton events were mapped analogously. For example, Left-Click and Drag of the mouse moved the Left Hand of the Skeleton and the 'S' key produced the "Dive" event and adjusted the skeleton accordingly.

6.2.1.2 Skelton Processing

The skeleton events produced by the hardware generators are passed to a series of skeleton processors. The processors each apply a specific algorithm to the incoming skeletons and return the resulting skeleton. These may be strung together to produce arbitrary processing strings.

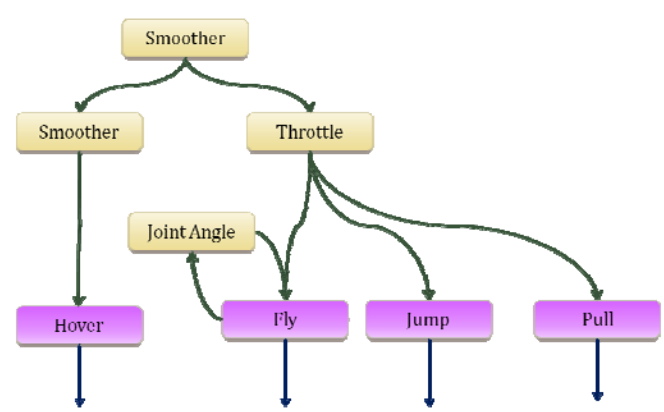


Figure 6-4 – Skeleton events are smoother and throttled then passed to gesture recognition processors.

[6] http://www.openni.org/

[7] For example PrimeSense http://www.primesense.com/, and ASUS http://usa.asus.com/

The processing for Tadpoles is shown in Figure 6-4. The suite of configurable skeleton processors developed that include:

- Smoother – which passes the skeleton joint positions through a low pass filter,
- Throttler – which reduces the data rate to no more than a specified rate, for example 4 per second,
- JointAngle – which calculates the angle for three specified joints,
- Differencer – which calculates the skeleton to skeleton change, and

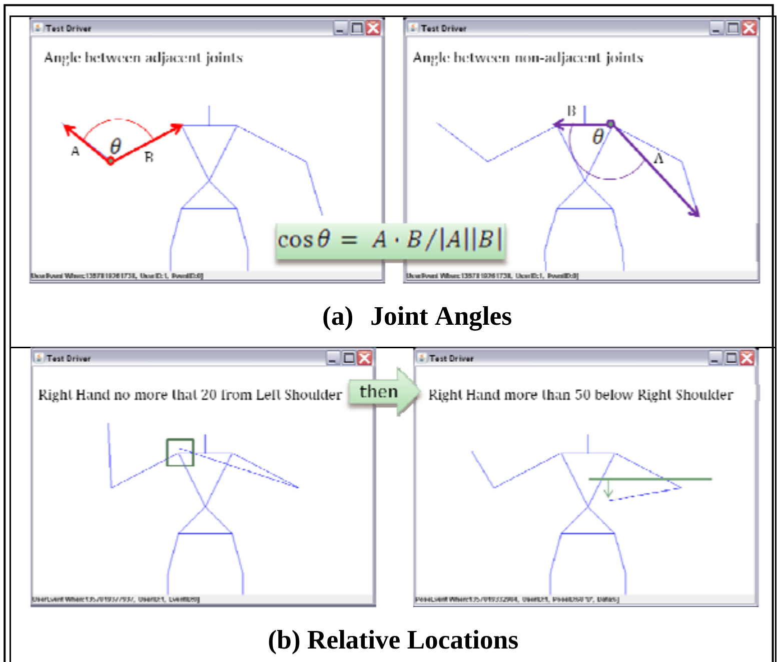
6.2.1.3 Gesture Recognition

The processed skeletons are passed to pose and gesture recognition processes which each look for specific position and motions in the skeleton. The recognition uses a finite state machine and transition event detectors to provide a flexible architecture for defining and recognizing gestures. One recognizer was used for each of the Tadpoles' input gestures.

6.2.1.4 Calibration

It was expected that a calibration of real-world to virtual world would be needed to accommodate the various physical sizes of students. However, it was found that a specific calibration was not needed. Through the combination of several methods, the needed mapping of real-world to virtual-world was accomplished on the fly during gameplay.

The first method was based in the projection of real world coordinates into virtual world coordinates. We found that a simple projection of the input devices 3D coordinates and domain into a common 2D screen coordinates with a fixed range of 640 by 480, essentially providing a "window" looking into the virtual world, allowed all inputs to be handled consistently. The change in the projection caused by player rotation, that is turning to face slightly away from the monitor, was not found to be significant enough to



(a) Joint Angles

(b) Relative Locations

Figure 6-5 -Scale invariant features used for pose and gesture recognition.

affect gameplay. However, for game scenarios which depend on motions in three dimensions, this should be reconsidered.

The second method used scale invariant features of the projected player's skeleton. Two types of invariant features were identified and used for Gesticulation: joint angles and relative locations. Joint angles used vector analysis to calculate the cosine of a skeleton joint using the formula shown in Figure 6-5(a). The cosine domain of +1 to -1 was then mapped to the required game inputs. For example, the illustrated angels were both alternative used to provide input to the "Flying" gesture in Tadpoles. Relative position simply looks for one joint to be in a specified position relative to another joint, as shown in Figure 6-5(b), which illustrates the "Pull Chute" gesture.

The final method was to provide visual feedback to the player which allows them to adjust their real world position as needed based upon real time feedback from the virtual world. This was used primarily for Deictic and Communicative control, for example menu selection. The identification of the desired choice used a large, obvious hand cursor which tracked the players right hand and provided visual cues indicating the object that is identified. The Communicative event to actually select the identified choice was a simple "Hover" defined as the cursor not moving beyond a specified distance over a specified time.

6.2.2 Game Environment

The Game Environment is the Unity 3D Integrated Development Environment[8]. It is an integrated development environment for the development of 3D video games or other real-time 3D interactive applications. The Unity development environment will run on both Microsoft Windows and Mac OS X platforms; the deployment environments include Windows, Mac, Xbox 360, PlayStation 3, handheld and web environments. PhysicsFun development used Unity to develop and deploy a standalone application for the Windows environment. Unity integrates the Input Framework using Microsoft .NET dll libraries provided by the framework.

7 Future Directions

The PhysicsFun4K24 project has produced a foundation for the development of immersive educational games. The seedling game itself has several potential directions in which it may be taken:

1. The seedling hardware suite may be used at conferences and expositions to present as an example to garner support for further development and generally encourage particularly the elementary students which are its target audience to pursue STEM educations,

[8] http://unity3d.com/

2. Because Unity 3D can produce executables for Web, iOS and Android[9], and the Input Framework is environment agnostic, the Tadpoles game should be readily deployable into other environments, with minor adaptations. For example:
    a. Web based – the keyboard and mouse interface allows Tadpoles to be deployed as a web application. While not have the immersive interfaces, the game would be much more widely accessible and could be incorporated into other web-based environment, such as the Immersive-3D Cyber STEM Academy[10],
    b. Mobile App – Both iOS and Android based hand held devices provide an accelerometer and gyroscope capability, which when combine with touch, could provide a very capable interface for Tadpoles.

In terms of further research and development of the immersive STEM methodology and this virtual environment, the next steps should be:

1. Hands on evaluation with students. To truly understand the effectiveness of this approach to STEM education, the environment needs to be presented to the students for which that it was intended. A small test with a limited number of students would provide directions for enhancements and refinement of both the curriculum presentation and gameplay aspects of Tadpoles. The Tadpoles game could be used as is, but better metrics logging and in game assessments are needed to provide qualitative information of analysis.
2. Explore new technology for gestures and immersive environments. Technology advances at a rapid rate; there have been several advances in immersive capabilities since the PhysicsFun project started. The exploration and incorporation of two capabilities could be especially impactful to the immersion and gameplay experience:
    - Use of a 3D capable projection video system[11] would present a tenfold increase in the size and visual impact of the virtual world, and
    - Utilization of multiple Kinect or Xtion devices to provide digit recognition[12] which would allow more effective us of hand base gestures and a richer interface capability.

[9] Unity 3D based games can be deployed to iOS, Android, MS Windows, Mac OSX, Flash, Linux, Web, PS3, Xbox and WiiU

[10] See http://www.immersive-3d.com/index.php

[11] 3D capable LDP projectors have become widely available, see http://www.projectorreviews.com/3d-projectors.php for an overview.

[12] See for example http://www.threegear.com/

3. Expand the curriculum. The limited curriculum used in Tadpoles showed proof of concept, but to complete and refine the curriculum to game methodology, and to provide a more interesting gameplay experience, additional content needs to be developed. There are a number of straight forward possibilities, keeping with a Navy Leap Frogs and kinematics theme, for example:
   - Diving underwater, replacing air drag with buoyancy,
   - A submarine picks up tadpoles using sonar to locate them, or
   - Preparations for a mission, loading equipment using ramps and pulleys.

8 Works Cited

A taxonomy of Gestures in Human Computer Interaction. **Karam, Maria and Schraefel, M.C. 2005.** s.l. : ACM Transactions on Computer-Human Interactions, 2005 йил.

Corti, Kevin. 2012. *PIXELearning.* [Online] 2012 йил. http://www.pixelearning.com.

Ferguson, Johnathan. 2012. *Interaction Designer.* Provo Utah : EduMetrics Institute http://www.edumetrics.org, 2012.

Karam, Maria. November 2006. *A framework for research and design of gesture-based human computer interactions.* s.l. : UNIVERSITY OF SOUTHAMPTON, November 2006.

2010. Sensor Fusion on Android Devices: A Revolution in Motion Processing. *YouTube.* [Online] August 08, 2010. http://youtu.be/C7JQ7Rpwn2k.

User-Defined Gestures for Surface Computing. **Jacob O. Wobbrock, Meredith Ringel Morris, Andrew D. Wilson. April 2009.** s.l. : CHI 2009 - ACM, April 2009. 978-1-60558-246.